**Faculdade de Engenharia da Universidade do Porto**



**Tema 2: Agência de Viagens**

**CAL 2ºano – 2MIEIC03 – Grupo H:**

Prof. Rosaldo Rossetti

**Estudantes & Autores:**

Gil Dinis Magalhães Teixeira 201505735@fe.up.pt

Ricardo José Santos Pereira 201503716@fe.up.pt

Rúben José da Silva Torres 201405612@fe.up.pt

07/04/2017

**Descrição do tema**

Com este trabalho pretendeu-se implementar um sistema que auxilie uma agência de viagens a conseguir a melhor solução para o cliente.

Deste modo, a agência trata da viagem e do alojamento do cliente (opcional).

As viagens entre as várias cidades podem ser feitas de vários meios de transporte.

Enquanto que o preço da viagem é sempre fixo, o preço do alojamento varia de acordo com o dia da semana e se houver festividade local.

Assim, a agência possui uma seleção de cidades à escolha e diferentes formas de viajar para cada uma, e de acordo com a seleção do cliente, tem como objetivo selecionar aquela que seja mais benéfica ao cliente em termos financeiros.

**Identificação e formalização do problema**

**-Dados de Entrada:**

-Cidade em que pretende iniciar a viagem;

-Cidade(s) que pretende visitar;

-Data em que pretende iniciar a viagem;

-Número de dias de alojamento(opcional);

-Transportes que não quer utilizar;

**-Restrições:**

-Destino e origem têm de ser diferentes;

-Data tem de ser válida;

**-Função Objetivo:**

-Calcular a melhor opção para o Cliente em termos financeiros, ou seja, selecionar entre as várias formas de transporte e alojamento selecionadas pelo cliente a mais acessível em termos financeiros.

**-Resultados Esperados:**

-Transportes a usar com a indicação da origem, destino, tipo de transporte, preço e duração de todas as viagens e o preço do alojamento.

**Solução Implementada**

**-Classes Usadas**

A agência de Viagens foi implementada usando a Classe **TravelAgency** que contém o Grafo *travelAgencyGraph* e vetor de strings *transportTypes*.

O Grafo é composto por vértices e arestas a unir os mesmos.

A informação dos vértices é representada pela classe **Node**, e a das arestas pela classe **Weight**.

A classe **Node** é composto pelo *nodeID* (identificador do nó), *cityName* (nome da cidade que representa), *accomodation*(alojamento) e *latitude* e *longitude* da cidade.

A classe **Accommodation** é composto pela *price* (preço base), vetor *datesPriceChange* (vetor de pares de datas em que o preço do alojamento é alterado), *percentageToChange*(vetor que representa a percentagem em que preço é alterado relativamente à da data) e *percentageDay* (map que contém as constantes que são multiplicadas pelo preço base de acordo com o dia da semana).

A classe **Weight** é composta por um vetor *allTrips*, composto por elementos da classe **TripInfo**.

A classe **TripInfo** é composto por *type* (o tipo de transporte usado. Ex: Avião), *price\_in\_euros* (preço da viagem) e *time\_in\_minutes* (tempo da viagem).

Além das classes acima descritas, é usada, como auxiliadora, a classe *Data*, que tem como membros: *ano*, *mes* e *dia.*

**-Funcionamento do programa**

O programa começa por ler 2 ficheiros “Nodes.txt” e “Edges.txt”. Se for encontrada alguma irregularidade é lançada uma exceção, se não a informação é armazenada no grafo *travelAgencyGraph.*

Depois, é chamada a função *mainMenu*, que mostra ao utilizador, na forma de um menu, as funcionalidades do programa.

**-Principais algoritmos implementados**

**-Variação do algoritmo de Gauss para a determinação do dia da semana:**

W=(d + floor(2.6\*m-0.2)+y+y/4+c/4-2\*s) mod 7,

Onde,

-d é o dia da semana;

-m é o número do mês – 2, exceto jan = 11 e fev = 12

-s é o século.

-y é são os 2 últimos algarismos do ano

-W é o dia da semana em que Domingo = 0 e Sábado = 6

- Cálculo do shortest Path:??

**-Ficheiros de Texto**

Os 4 ficheiros de texto começam com um cabeçalho a indicar o tipo de ficheiro (Exemplo: “#Edges\_start”, “#Vertex\_start” e terminam com o rodapé “#End”.

O ficheiro de Arestas guarda a informação na forma “Id da Aresta ; Id do nó Origem ; id do nó Destino ; Tipo de Transporte ; preço da viagem ; duração da viagem em Minutos”, sendo que os últimos 3 elementos podem ser repetidos. Exemplo: “7;4;1;Ship;5;180;Airplane;50;100”.

O ficheiro de Nós guarda a informação na forma “Id da Nó ; Nome da cidade; preço base de alojamento ; longitude ; latitude ; data de inicia de época com preço diferente ; data de fim de época ; percentagem de alteração de preço”, sendo que os últimos 3 elementos são opcionais e podem ser repetidos. Exemplo “1;Porto;30;-8.610243;41.149968;1/12;28/2;0.8;1/6;1/7;1.0;1/7;1/9;1.3”

**Diagramas de classes (UML)**

**Listas de Casos de Utilização**

 Criação de uma Agência de Viagens.

 Leitura de ficheiros de dados.









**Principais Dificuldades**

As principais dificuldades na elaboração do trabalho solicitado consistiram basicamente na programação de um algoritmo ………………………..

Mesmo assim, estamos convictos de que realizamos com acerto a proposta apresentada.

**Esforço dedicado por cada elemento do grupo**

A dedicação ao trabalho pelo par de elementos que constituem o grupo foi equitativa, pelo que é justo reconhecer o equilíbrio o esforço despendido.

**Referências bibliográficas**

1. **Cheriton School of Computer Science - University of Waterloo, 1998 “****How to determine the day of the week, given the month, day and year”.** Disponível em: <https://cs.uwaterloo.ca/~alopez-o/math-faq/node73.html>. Data de acesso: 1 de março de 2017.
2. **Wikipedia, 2017, “Determination of the day of the week”.** Disponível em: <https://en.wikipedia.org/wiki/Determination_of_the_day_of_the_week>. Data de acesso: 1 de março de 2017.